

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

## **IMAGES ARE BEST AVAILABLE COPY.**

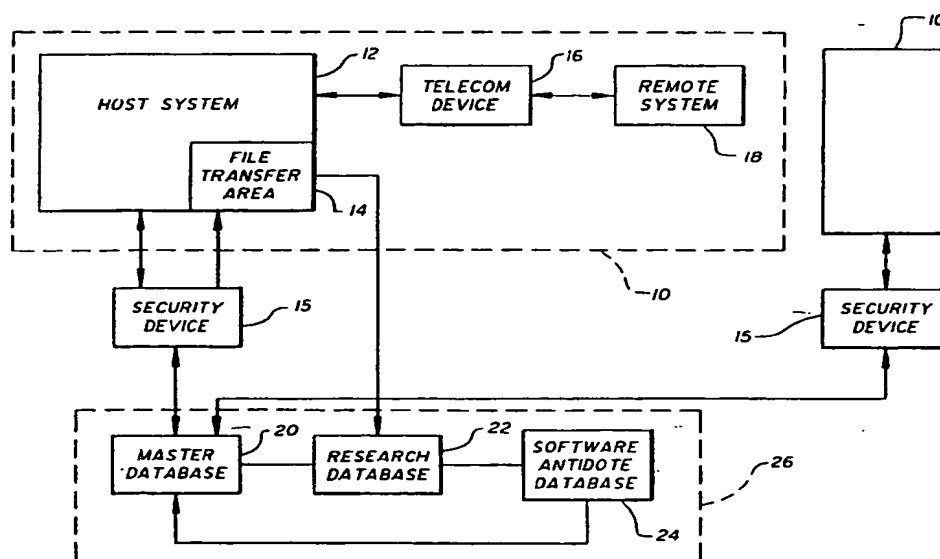
As rescanning documents *will not* correct images,  
Please do not report the images to the  
Image Problem Mailbox.

**This Page Blank (uspto)**



## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<b>(51) International Patent Classification <sup>5</sup> :</b>  <b>H04L 9/00</b>	<b>A1</b>	<b>(11) International Publication Number:</b> <b>WO 93/25024</b>  <b>(43) International Publication Date:</b> 9 December 1993 (09.12.93)
<b>(21) International Application Number:</b> PCT/US93/05029 <b>(22) International Filing Date:</b> 26 May 1993 (26.05.93)  <b>(30) Priority data:</b> 07/888,909                      26 May 1992 (26.05.92)                      US  <b>(71) Applicant:</b> CYBERLOCK DATA INTELLIGENCE, INC. [US/US]; 906 Woodbrook Lane, Philadelphia, PA 19150 (US).  <b>(72) Inventor:</b> BRANHAM, Reginald, K. ; 906 Woodbrook Lane, Philadelphia, PA 19150 (US).  <b>(74) Agent:</b> HEIDELBERGER, Louis, M.; Reed Smith Shaw & McClay, 2500 One Liberty Place, 1650 Market Street, Philadelphia, PA 19103-7301 (US).		<b>(81) Designated States:</b> CA, JP, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).  <b>Published</b> <i>With international search report.</i>

**(54) Title:** COMPUTER VIRUS MONITORING SYSTEM**(57) Abstract**

A method and an apparatus for preventing the infection of computer systems (10) by computer viruses is disclosed. The computer virus monitoring system provides an external security device (15) that stores copies of the host boot sector and file allocation table, and electronic fingerprints of executable files on the host system disk. The external security device (15) monitors writes to the host disk and informs a network monitoring host (26) if the boot sector, file allocation table or electronic fingerprints are altered. The network monitoring host (26) responds to such an emergency by saving the status of the host system (12) and transferring corrective software to remove the virus. The computer virus monitoring system further provides for periodic scanning of the host files to detect and eradicate known viruses.

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FR	France	MR	Mauritania
AU	Australia	GA	Gabon	MW	Malawi
BB	Barbados	GB	United Kingdom	NL	Netherlands
BE	Belgium	GN	Guinea	NO	Norway
BF	Burkina Faso	GR	Greece	NZ	New Zealand
BG	Bulgaria	HU	Hungary	PL	Poland
BJ	Benin	IE	Ireland	PT	Portugal
BR	Brazil	IT	Italy	RO	Romania
CA	Canada	JP	Japan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SK	Slovak Republic
CI	Côte d'Ivoire	LJ	Liechtenstein	SN	Senegal
CM	Cameroon	LK	Sri Lanka	SU	Soviet Union
CS	Czechoslovakia	LU	Luxembourg	TD	Chad
CZ	Czech Republic	MC	Monaco	TG	Togo
DE	Germany	MG	Madagascar	UA	Ukraine
DK	Denmark	ML	Mali	US	United States of America
ES	Spain	MN	Mongolia	VN	Viet Nam
FI	Finland				

## COMPUTER VIRUS MONITORING SYSTEM

### Field Of The Invention

This invention relates to detection of viruses or logic bombs in computer systems and for taking responsive action upon such detection.

### Background Of The Invention

Security of computer systems is becoming increasingly important. Computer viruses and logic bombs ("viruses") are proliferating, and their potential impact is increasing with the increasing use of network systems. Efforts to address the problems posed by viruses have included backing up the contents of computer systems on magnetic tape or other media, and storing the backup in a safe location. Such systems are time consuming and expensive to implement, and only serve to mitigate the effects of a problem after a problem has occurred. Others have developed software programs, resident on a computer system to be protected, to monitor the system for viruses. Such protection systems have certain drawbacks: for instance, because they are resident on a system which may become infected, they are similarly subject to disruption or corruption. Such systems may also be cumbersome to update to respond to newly encountered viruses.

### Summary Of The Invention

It is therefore an object of the invention to provide an improved computer virus monitoring system. In accordance with the invention, the system includes a monitoring computer which is coupled to the system to be protected. The monitoring computer monitors the boot sector and file allocation table of the disk drives of the

-2-

host system to be protected, and creates and maintains identification data for each executable file. The monitor detects attempts to alter these files or attempts by a file to alter itself or another file and takes appropriate responsive action to protect the host system from unauthorized alteration. The monitor also includes means for scanning the host system to detect particular viruses, and to take appropriate protective action upon detection of a virus. In a particularly preferred embodiment, such monitors on a number of host systems are coupled by a communication network to facilitate analysis of viruses and distribution of antidotes therefor.

Other objects and features of the invention will be understood with reference to the following specification and claims and the drawings.

#### Brief Description Of The Drawings

Figure 1 is a block diagram showing the basic elements of a system in accordance with the invention.

Figure 2 is a flow diagram illustrating a virus protection method which may be implemented on the system of Figure 1.

Figure 3 is a flow diagram of a method of operating the host system of Figure 1 in accordance with the invention.

Figure 4 is a flow diagram illustrating a method of operation of the monitor of Figure 1 in accordance with the invention.

Figure 5 is a flow diagram illustrating a method of operation of the monitor in response to an emergency condition.

Figure 6 is a flow diagram illustrating a method of communication between monitoring system at a number of host locations and a central monitoring system.

-3-

Figure 7 is a block diagram illustrating the basic elements of a preferred embodiment of a security device in accordance with the invention.

#### Detailed Description of the Invention

5           Figure 1 is a diagram showing the functional elements of the preferred embodiment of the system of the present invention. A computer network 10 which is to be protected (the "protected network") includes a host system 12 operating at a customer field site that communicates  
10 with one or more remote systems and communication nodes 18 via one or more telecommunication devices 16. In a typical application, host system 12 may be an IBM fileserver, an Apple fileserver, or other computer system. Telecommunications device 16 may be a modem, a network  
15 card, or a satellite transceiver, for example.

Security device 15 monitors the operation of host system 12 in three different modes to automatically detect viruses and prevent adverse effects. Security device 15 monitors the boot sector and file allocation  
20 table (FAT) of disk drives in host system 12. It creates and maintains identifying data as a "fingerprint" for each executable file. These files should never be altered; therefore, if an attempted alteration is detected, appropriate protective action may be taken. Finally,  
25 security device 15 scans host system 12 periodically to detect particular viruses.

Security device 15 is coupled to host 12 via a parallel connection to the standard architectural bus of host 12 in the preferred embodiment; however, security  
30 device 15 may be coupled to host 12 via a serial port in an alternate embodiment. Security device 15 is also coupled to a security network monitoring host 26, by a telephone line 19 or other communications link.

Desirably, a number of protected networks 10,  
35 each with its own security device 15, are connected in a

-4-

network with a security network monitoring host computer system 26. Security network monitoring host 26 includes a master service database 20 to monitor standard operation of security devices 15. It also includes a research service database 22 which may be called by a protected network 10 in case of an emergency. Because research service database 22 includes storage media such as disk drives, it may become infected by a virus. Therefore, research database 22 is kept separate from the master service database 20. Security network monitoring host system 26 also includes an antidote software service database 24, in which the characteristics of newly encountered viruses are stored and in which solutions to eradicate them are maintained. Antidote software may be transferred via master database 20 to security devices 15 in the monitoring network as needed.

In a system such as that shown in Figure 1, it is important for security device 15 to be as secure and reliable as possible. Therefore, security device 15 is preferably a stand alone unit without a disk drive or any other alterable storage device except RAM, to minimize the possibility of infection. Security devices 15 are provided with battery backup to maintain operation during a power failure. An emergency routine is called when security device 15 detects tampering, such as an unauthorized attempt to open its housing. Communications between security network monitoring host 26 and security devices 15 or protected networks 10 are desirably protected by requiring device-specific electronic identification numbers (EINs) to initiate communication.

Figure 2 is a flow diagram showing a method for operating a host system 12 and a security device 15 to provide virus protection for executable files. In this method, an electronic binary "fingerprint" of identifying data is created and assigned for each executable file on host system disk drive pack. The identifying data is then after each write access monitored by security device 15.



-5-

If the identifying data changes, then security device 15 will halt the protected system and initiate a call to the research database 22 at security network monitoring host 26.

5           The process of Figure 2 for monitoring the software of the protected system starts in step 50 and new software is added to the protected system in step 52. In step 54 the fingerprint, which can be conventionally generated as a check sum, a cyclic redundancy check (CRC) or other error detecting coding scheme, is created to provide an identifying code that is unique to the file. In a preferred embodiment, the fingerprint is generated by host system 12, although in alternate embodiments the fingerprint could be generated on security device 15 or on a dedicated circuit card. The fingerprint is saved on host system 12 in step 58, and on security device 15 in step 56. Steps 60 and 62 provide an ongoing process, for determining the status of the software by monitoring the fingerprint. The host fingerprint is read in step 60. In step 62 the host fingerprint is compared with the corresponding fingerprint in security device 15. If the host fingerprint has not changed, the process returns from step 62 to step 60 to continue the monitoring process. If the fingerprint has changed, then in step 64 security device 15 halts operation of host system 12. Security device 15 then initiates a call to the system operator of host system 12 in step 68, and in the preferred embodiment, a call to the research database 22 of security network monitoring host 26 in step 70. Also in response to detection in step 62 of a fingerprint change, security device 15 attempts in step 72 to destroy the virus which caused the fingerprint change.

          The process of destroying the virus involves comparison of the contents of files stored on host 12 with characteristics of known viruses stored in security device 15 downloaded from master database 20 of network host 26. If characteristics of a known virus are detected, antidote

-6-

software, stored in antidote software database 24, is executed to destroy the virus in step 74. Upon successful detection and destruction of a virus, a report of this activity is generated in step 76, transmitted to the  
5 research database 22 of security network monitoring host 26 in step 78, and transmitted to the system operator of host system 12 in step 80. Step 82 returns the process to step 54, in which a new fingerprint is created, and then saved. Host system 12 may thereafter continue its  
10 operation free from the virus. If in step 74 the virus has not been successfully destroyed, a further call is initiated by security device 15 to research database 22 for further assistance in identification and destruction of the virus, while operation of host system 12 is  
15 disabled to prevent consequential damage from the virus.

Figure 3 is a flow diagram of the operation of host system 12 relating to security device 15. Initialization steps of host system 12 include porting the boot sector and FAT of host system 12 to security device  
20 15 in steps 90 and 92 respectively, creating a fingerprint file in step 94 and porting the fingerprint file to security device 15 in step 96. In the host monitoring process, security device 15 monitors write access signals in step 102, indicating an instruction to write to a disk.  
25 When a write access signal is received, host system 12 provides information to security device 15 to use in tests to verify system integrity and virus free input. In step 104, host system 12 sends a write access interrupt to security device 15. In step 106, security device 15 reads  
30 its interrupt status to determine whether the emergency interrupt flag is set. If an emergency interrupt is indicated, the keyboard is locked and host 12 is halted in step 108. In step 110 the write access is examined to determine whether it seeks to write to an existing  
35 executable file; if so, in step 112 the keyboard is locked and host system 12 is halted. Otherwise, in step 114 the write access is examined to determine whether it seeks to

-7-

write a new executable file or batch. If so, in step 116 host 12 calls a scan routine from security device 15 and executes it. The scan routine examines the file to be written for viruses and the like. If the scan indicates the presence of a virus or the like, step 118 directs the process to step 120, where an emergency interrupt is sent to security device 15, and step 122, where the keyboard is locked and host system 12 is halted. If in step 118 no viruses or other problems have been detected, or if step 114 has determined that the write access is not for a new executable file, the file may be written to disk. In step 124, host system 12 updates the FAT on security device 15, creates new fingerprints in step 126, and updates the fingerprint data on security device 15 in step 128.

Host system 12 also waits for interrupt from security device 15 to run tests to verify the entire system is virus-free. This is done to determine at what time any system errors occur to help with data recovery in case that becomes necessary. Thus, in step 130 if a fingerprint interrupt is received from security device 15, host system 12 in step 132 ports its fingerprint data to security device 15. If host system 12 then receives a fingerprint emergency interrupt from security device 15 in step 134, it locks the keyboard and halts host system 12 in step 136; otherwise, control is passed to step 138 and host system 12 returns to its start status in 100. If in step 140 a scan interrupt is received from security device 15, host system 12 calls the scan routine from security device 15 and executes it in step 142. If no viruses or the like are detected in the scan, step 144 directs a return to the start status via step 138. If, however, the scan process results in the detection of a virus or the like, in step 146 an emergency interrupt is sent to security device 15 and host system 12 is halted and its keyboard locked in step 148.

Figure 4 is a flow diagram illustrating the operation of security device 15. Security device 15

-8-

initialization includes steps 150, 152, and 154, where boot sector, FAT, and fingerprints ("CRCs") of host 12 are copied and mapped into security device 15 memory.

The security device 15 monitoring process starts in step 156, and in step 158 security device 15 monitors the host until a write access interrupt is received from the host. In step 160, security device 15 compares the boot sector files stored in its memory with the boot sector files stored in host system 12. If these are not the same, step 162 directs the process to an emergency state in step 164. If the boot sector files are determined to be the same as in step 162, step 166 determines whether the write access seeks to write to a .EXE, .COM file, or the like. If so, security device 15 goes to an emergency state in step 168 and follows the emergency procedure shown in Figure 5. If not, step 170 determines whether the write access seeks to write a new executable file. If so, step 172 scans the file to be written to determine whether it contains any viruses. If the test is not passed, security device 15 is directed in step 174 to go to an emergency state in step 176. If the test in step 174 is passed, or if in step 170 the write access was not for a new executable file, the file may be written. In step 178, the FAT is updated on host system 12 and security device 15, and in step 180 fingerprint data for the file is created on the host 12 and updated fingerprint data is sent to security device 15.

In addition to monitoring initiated upon write accesses, security device 15 also automatically periodically monitors fingerprint data and scans for viruses. In step 182, if an internal clock in security device 15 indicates that it is time for a fingerprint check, in step 184 the fingerprint data stored in security device 15 is compared with the current fingerprint data of host system 12. If the fingerprint data is the same, step 194 directs the process to step 190 and a return to start

-9-

condition 154. If the fingerprint data is different, security device 15 goes to an emergency state in step 196.

In step 186, if security device 15 clock indicates that it is time to perform a scan, a scan is performed in step 188 to determine whether the contents of host system 12 contain any viruses or the like by comparison with identifying data stored in security device 15. If the scan detects no such code, step 192 directs the process to the start condition through step 190; if a virus is detected, security device 15 goes to an emergency state in step 198.

Figure 5 is a flow diagram illustrating emergency condition operation of security device 15, which starts at step 200. A security device 15 sends an emergency interrupt to host system 12 in step 202, instructing host system 12 in step 204 to close all files and return to the DOS prompt (or an equivalent prompt in a non-DOS system), and in step 206 to lock the keyboard and halt operation. This action isolates the virus in host system 12. Host system 12 remains locked until the proper unlock password is entered in step 208, in which event host system 12 will unlock the keyboard in step 210 and save the status to disk in step 212, display the status in step 214, and write the status to the printer in step 216.

In addition to sending an emergency interrupt to host system 12, in the preferred embodiment security device 15 is connected in a network, and in step 218 calls the monitoring host 26. In step 220, security device 15 identifies itself by sending a login code and the electronic identification number for the particular security device 15. If the electronic identification number is accepted, security device 15 will be permitted to login to the research database, a call is generated to the monitoring network system operator in step 222, and the status of the protected network 10 which generated the emergency condition is displayed in step 224. After the monitoring network system operator identifies and corrects

-10-

the problem which generated the emergency condition, the system operator resets security device 15 in step 222, and normal operation of protected network 10 resumes in step 228.

5           Figure 6 is a flow diagram illustrating the operation of network monitoring system 26. In step 250, network monitoring system 26 is responding to a connect attempt initiated by a security device 15, in step 218 of Figure 5. In step 252, network monitoring system 26  
10       prompts security device 15 for its EIN and login code. In step 220, security device 15 sends its logon code and EIN to network monitoring system 26. In step 254, network monitoring system 26 checks the received EIN and logon code to verify that the communication was initiated by a  
15       legitimate security device 15, not by a computer hacker. To ensure the security of the system, a preferred embodiment of the invention uses a 128 bit EIN. In a preferred embodiment, network monitoring host 26 waits only a limited amount of time, approximately 9 seconds, to  
20       receive logon information after it detects a carrier signal on the telephone link. The combination of a 128 bit EIN and a limited time to respond effectively eliminates the possibility of a hacker logging onto the system. If network monitoring system 26 detects an  
25       invalid EIN or a time-out condition, it disconnects the telephone hookup in step 256. If network monitoring system 26 detects a valid EIN in step 254, the logon procedure is executed in step 258.

          In step 260, network monitoring system 26  
30       determines whether or not the call is an emergency call. If the call is an emergency call, the status of host system 12 is saved in research database 22 in step 272, displayed in step 274 and printed out in step 276, so that the virus can be isolated. In parallel with steps 272,  
35       274 and 276, a call is generated by an emergency routine within host 12 to the system operator of network monitoring system 26 in step 278. In step 280, network

-11-

monitoring system 26 checks whether host system 12 has been reset. Once host system 12 has been reset, a logoff procedure is followed in step 282 and network monitoring system 26 disconnects telephone link 19 from security device 15 to research database 22 in step 284.

If, in step 260, network monitoring system 26 determines that the call is not an emergency call, it checks in step 262 whether the call is a report call. If the call is not a report call, the system operator is called in step 264 and network monitoring system 26 enters terminal mode. If the call is a report call, network monitoring system 26 creates a report file including the time and date in step 266, stores the report file in step 268 and prints the file in step 270. After the file is sent to the printer, network monitoring host 26 initiates a logoff procedure in step 282 and disconnects the telephone link from security device 15 in step 284.

Figure 7 is a block diagram of a preferred embodiment of security device 15. Security device 15 includes a processor 300 connected to a control bus 354, a data bus 356 and an address bus 358. Processor 300 communicates through buses 354, 356 and 358 to read only memory (ROM) 306, random access memory (RAM) 308, I/O interface 320, I/O interface 316 and liquid crystal display (LCD) driver 310. Processor 300 further includes an input from Clock 302 and from Reset Input 304. LCD driver 310 is coupled through LCD interface 312 to LCD 314. LCD 314 is used by security device 15 to display status information to a maintenance technician or an engineer, to aid in isolating a virus found in host system 12. Security device 15 includes ROM 306 and RAM 308 as storage devices. ROM 306 is utilized for storing the firmware which controls processor 300. RAM 308 is used by Processor 300 to store temporary information, including fingerprint files, boot sector information and file access table information. Security device 15 communicates with host 12 through I/O interface 320. I/O interface 320 is

-12-

preferably coupled to file transfer area 14 of host 12 via a parallel connection, although in another embodiment a serial port is utilized. Security device 15 is assigned a random address location in the memory space of host system 5 12, so that a Hacker cannot try to access RAM 308. Security device 15 is coupled to modem 318 through I/O interface 316. Security device 15 uses modem 318 to access the master database 20 of host of network monitoring system 26 through telephone line 19.

10           It is to be understood that the foregoing is merely illustrative of the principles of this invention, and that various modifications can be made by those skilled in the art without departing from the scope and spirit of the invention.



-13-

What Is Claimed Is:

1. A monitoring system for detecting software viruses in a host computer system having a host memory and a file storage memory; for storing an executable software file by means of a write access, said monitoring system comprising:

means for generating an electronic fingerprint of the executable software file when the file is written to the file storage memory by means of a first write access;

a security device coupled to the host computer, said security device comprising means for monitoring write accesses of the file storage memory, fingerprint memory means for storing a copy of the fingerprint and emergency interrupt response means for halting and locking the host computer in response to an emergency interrupt;

means for transferring a first copy of the fingerprint to the fingerprint memory of the security device and for transferring a second copy of the fingerprint to the host memory; and

comparison means for comparing the first copy of the fingerprint stored in fingerprint memory with the second copy of the fingerprint stored in the host memory when a subsequent write access of the executable software file occurs and for generating an emergency interrupt to the security device if the first and second copies are not identical.

2. The system of claim 1, wherein the host computer system includes the fingerprint generation means.

-14-

3. The system of claim 1, wherein the security device includes the fingerprint generation means.

4. The system of claim 2, wherein the electronic fingerprint is a checksum code.

5. The system of claim 2, wherein the electronic fingerprint is a cyclic redundancy check code.

6. The system of claim 1, wherein the host computer includes the comparison means.

7. The system of claim 1, wherein the security device includes the comparison means.

8. A monitoring system for detecting software viruses in a protected network including a host computer coupled to a remote computer system, the host computer having a host memory and a host file storage memory for storing a software file by means of a write access, the monitoring system comprising:

means for generating an electronic fingerprint of the software file when the file is written to the file storage memory;

a security device coupled to the host computer, said security device comprising means for monitoring write accesses of the file storage memory, fingerprint memory means for storing a copy of the fingerprint and emergency interrupt response means for halting and locking the host computer in response to an emergency interrupt;

means for transferring a first copy of the fingerprint to the fingerprint memory of the security device and for transferring a second copy of the fingerprint to the host memory; and

comparison means for comparing the first copy of the fingerprint stored in fingerprint memory with the second copy of the fingerprint stored in the host memory when a subsequent write access of the

-15-

software file occurs and for generating an emergency interrupt to the security device if the first and second copies are not identical.

5           9. The system of claim 8, further comprising:  
              means, in the security device, for  
              storing a copy of the file allocation table  
              of the host computer system; and

10               comparison means responsive to a write  
              access of file storage memory, for  
              comparing the copy of the file allocation  
              table stored in the security device memory  
              means with the file allocation table stored  
15               in the host system when a write access  
              occurs and for generating an emergency if  
              the copy of the file access table is not  
              identical to the file allocation table  
              stored in the host system.

20           10. The system of claim 9, further comprising:  
              means for copying the boot sector of  
              the host computer system to the memory  
              means in the security device; and

25               comparison means for comparing the  
              copy of the boot sector stored in the  
              security device memory with the boot sector  
              stored in the host memory when a write  
              access occurs and for generating an  
              emergency interrupt if the copy of the boot  
30               sector is not identical to the boot sector  
              of the host system.

11. The system of claim 8, further comprising a  
network monitoring system coupled to the security device  
and to the host system, for monitoring a plurality of  
security devices.

35           12. The system of claim 11, wherein the network  
monitoring system further comprises a research database  
for compiling performance characteristics for each

-16-

security device and protected network, and an antidote software database for storing software to overcome known computer viruses.

13. The system of claim 12, wherein the  
5 research database contains known computer virus code patterns.

14. The system of claim 13, further comprising means for scanning the host file storage memory to detect the known software viruses stored in the research  
10 database.

15. A method for protecting a host computer system including a file storage memory from infection by a software virus, comprising the steps of:

15 generating an electronic fingerprint for an executable software file being stored in the file storage memory;

storing a first copy of the electronic fingerprint in the host memory and storing a second copy of the electronic fingerprint  
20 in an external security device;

monitoring write accesses to the file storage memory;

25 comparing the first and second copies of the fingerprint whenever the write access occurs; and

halting and locking the host computer when the first and second copies of the fingerprint differ, indicating the possible presence of a virus, so that the virus  
30 cannot damage the host computer system.

16. The method of claim 15, further comprising the steps of:

35 transferring a copy of the boot sector of the host computer system to the security device memory;

periodically comparing the copy of the boot sector with the host boot sector to

-17-

determine whether the host boot sector has changed; and

if the host boot selector has changed, halting and locking the host computer system to prevent possible damage by a virus.

17. The method of claim 16 further comprising the steps of:

transferring a copy of the file allocation table of the host computer system to the security device memory:

periodically comparing the file allocation table with the copy of the file allocation table stored in the security device memory to determine whether its fingerprint has changed; and

if the file allocation table has changed, halting and locking the host computer system to prevent possible damage by a virus.

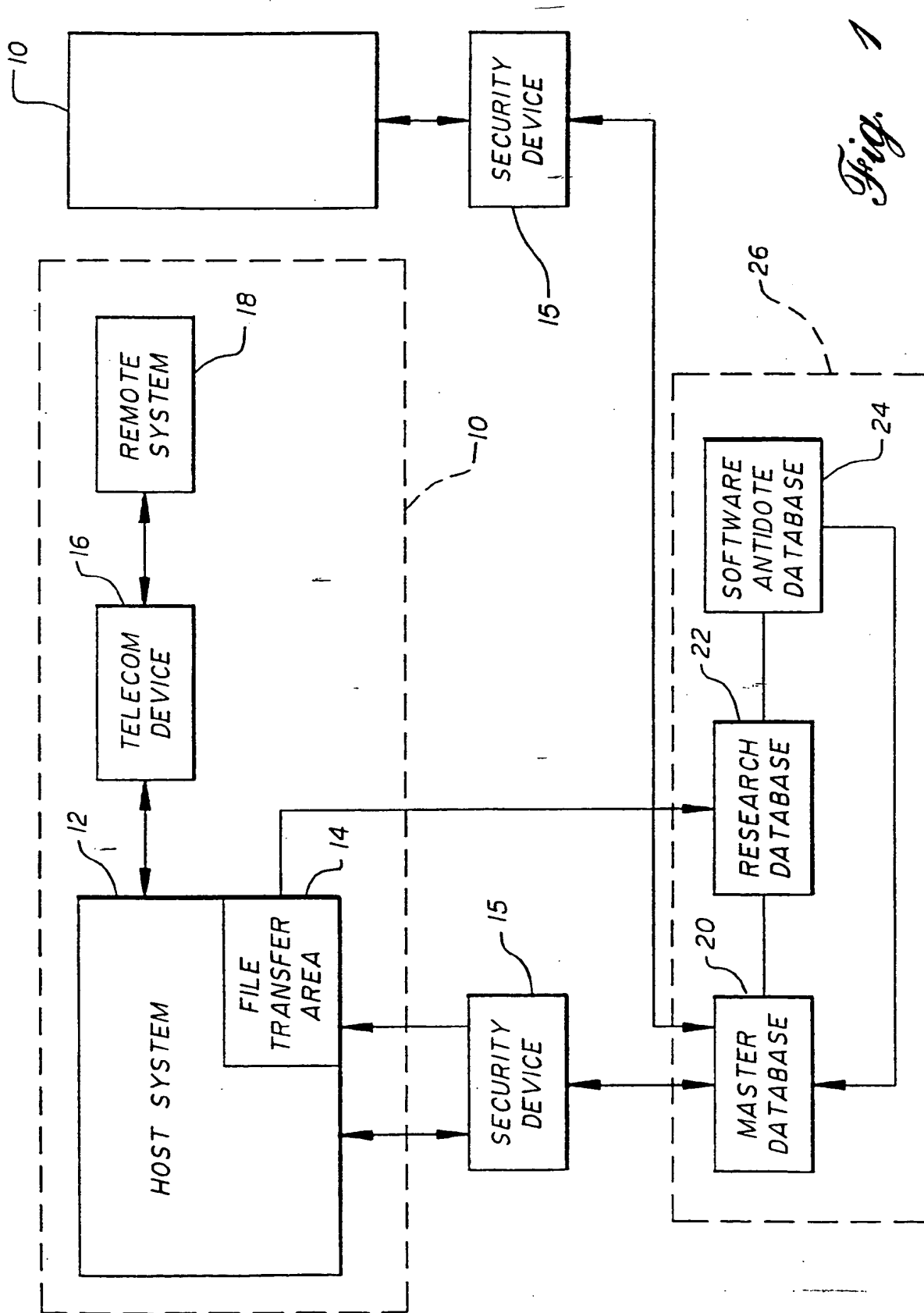
18. The method of claim 17, further comprising the steps of:

storing known computer virus code patterns in a first database;

storing antidote software for said known computer virus code patterns in a second database;

periodically scanning the file storage memory of the host computer system for said known computer virus code patterns; and

halting the host computer system and executing said antidote software if a virus is detected.



*Fig. 1*

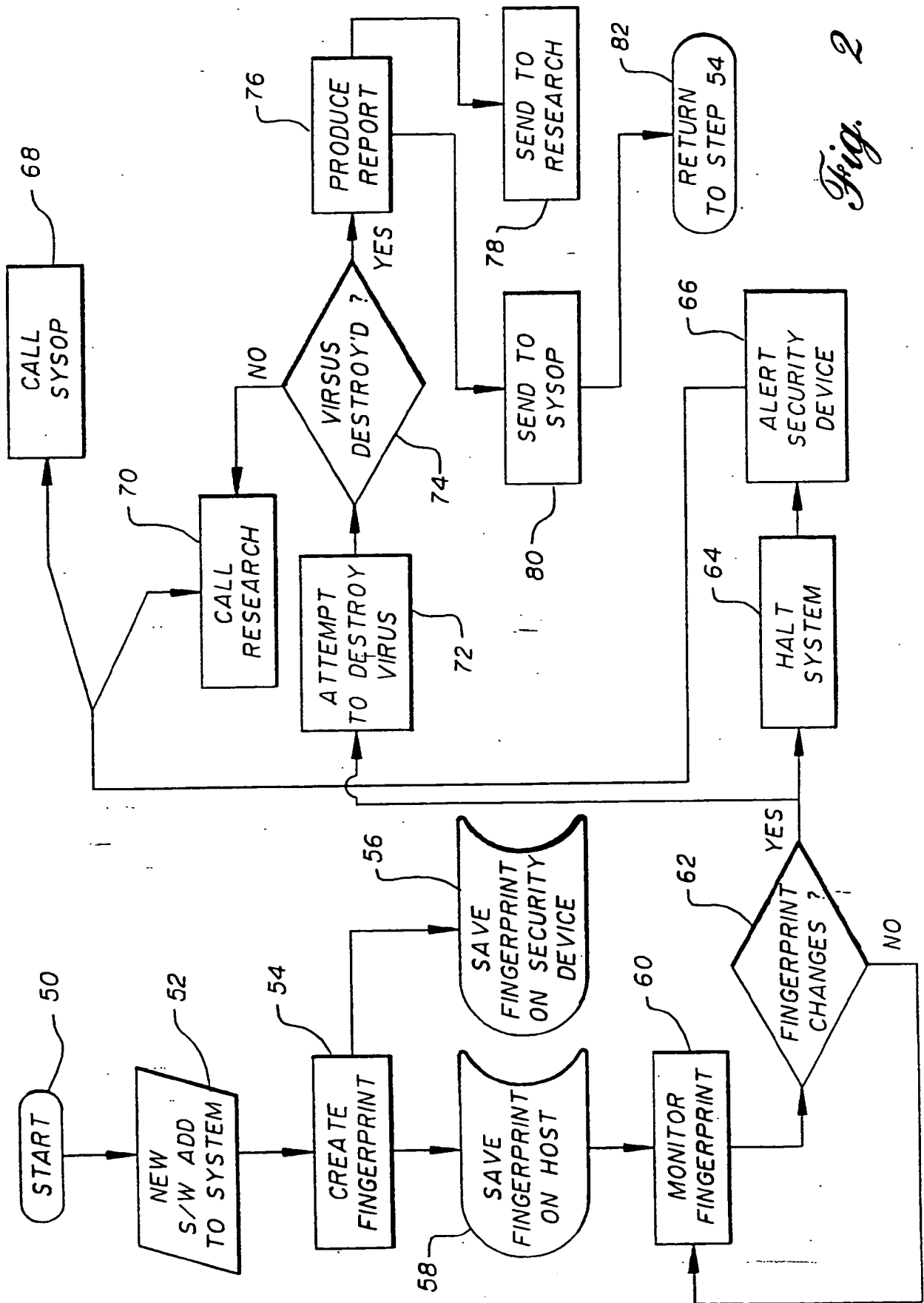
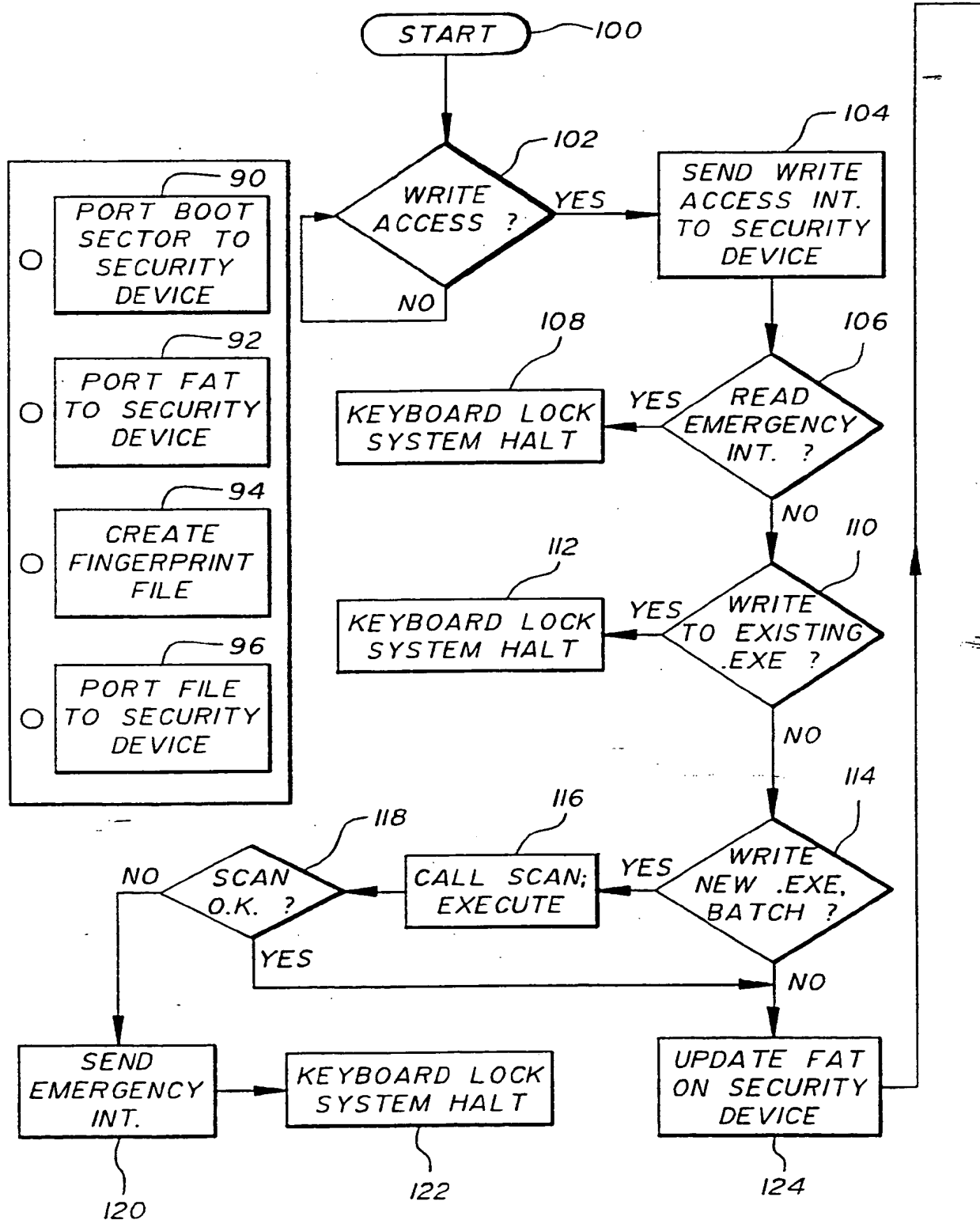
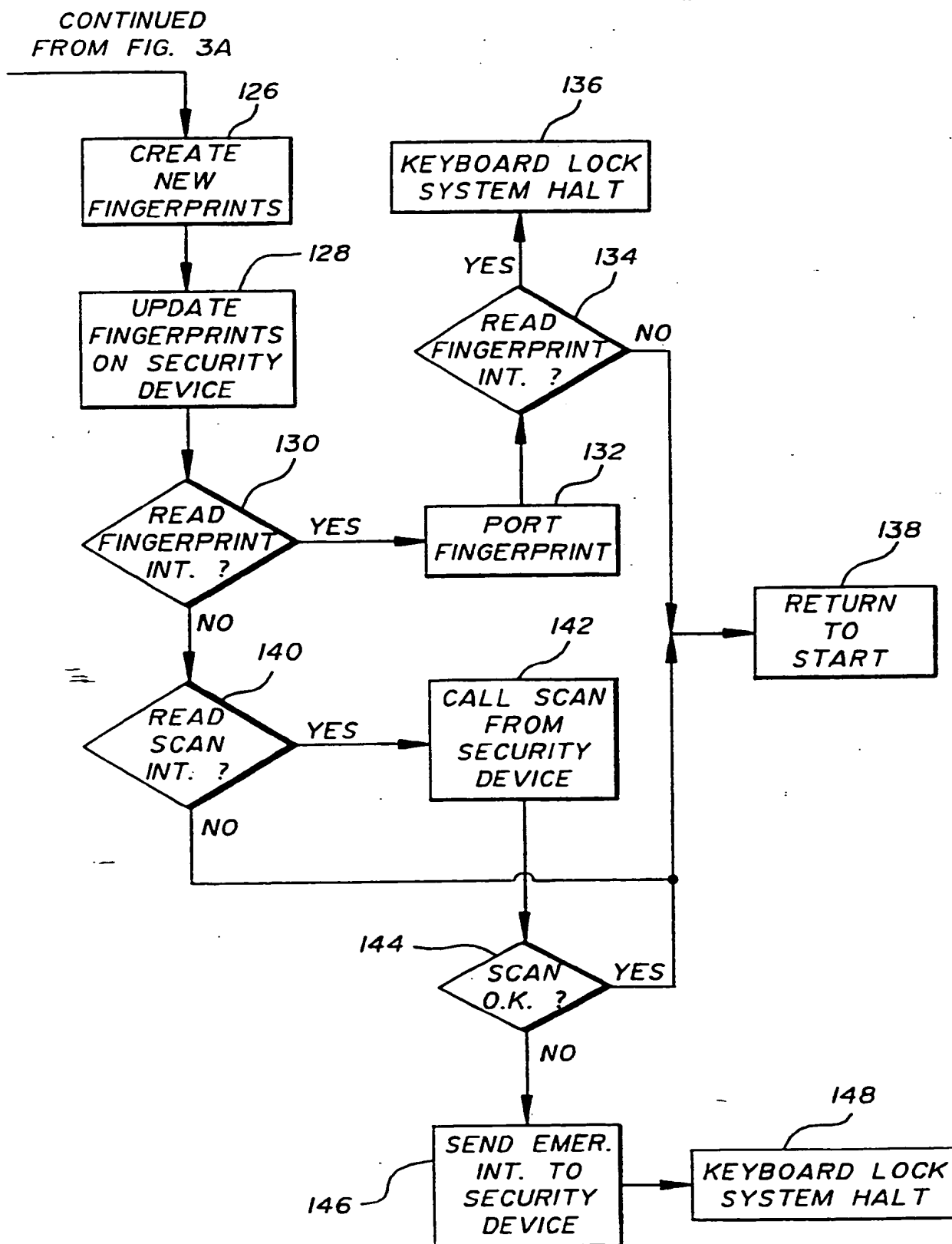


Fig. 2

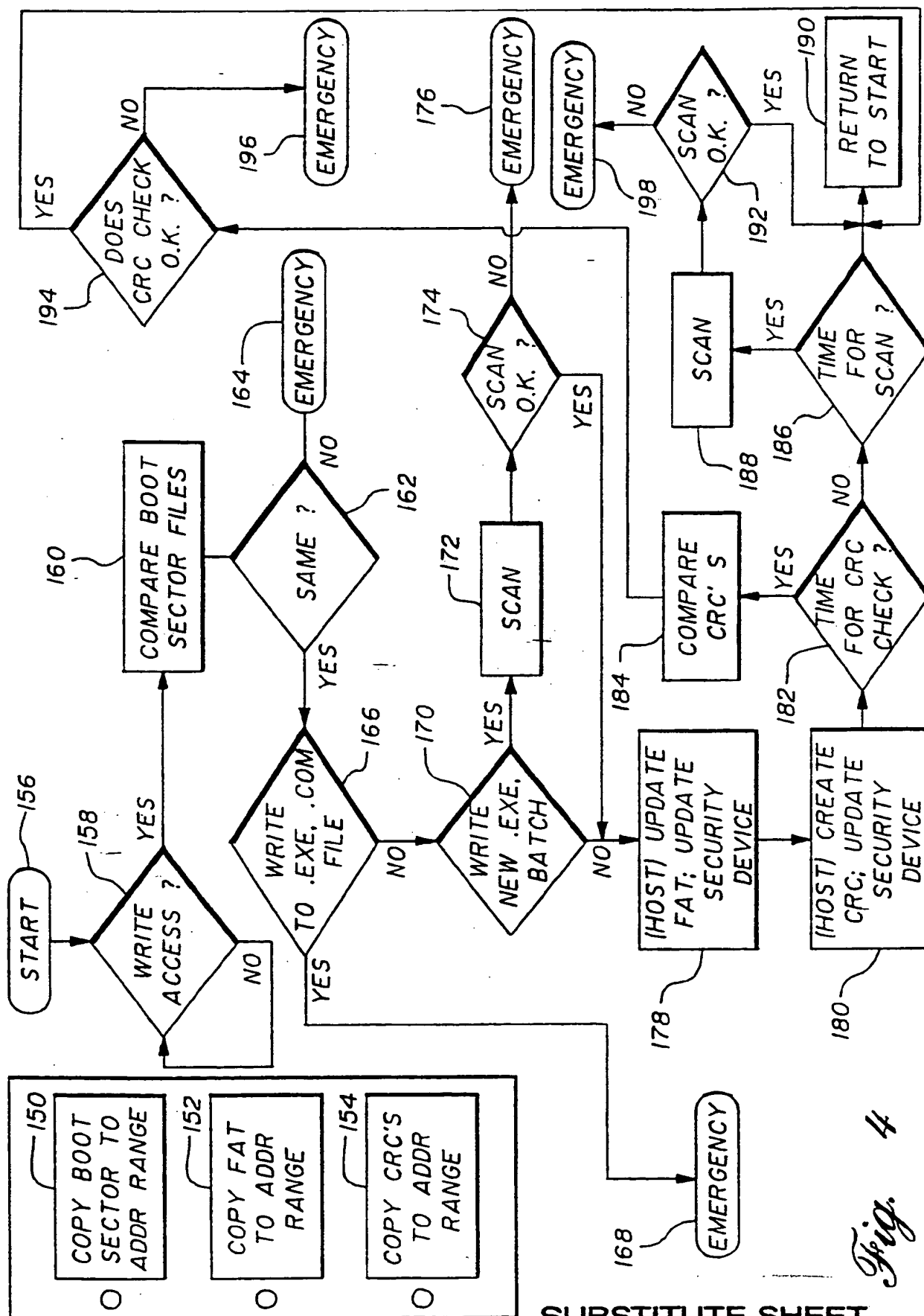
CONTINUED  
ON FIG. 3B

*Fig. 3A*  
SUBSTITUTE SHEET



*Fig. 3B*

SUBSTITUTE SHEET



# SUBSTITUTE SHEET

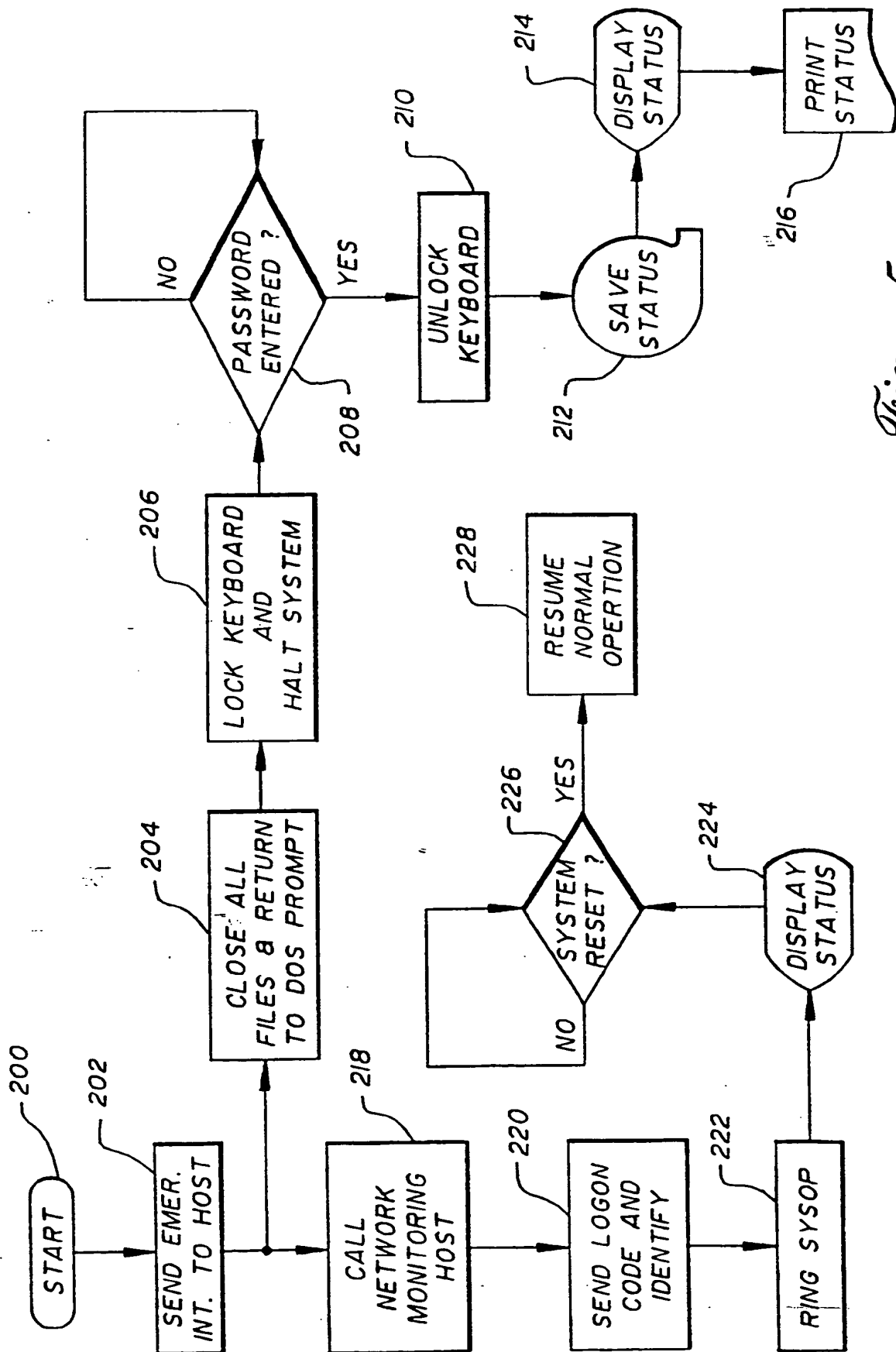
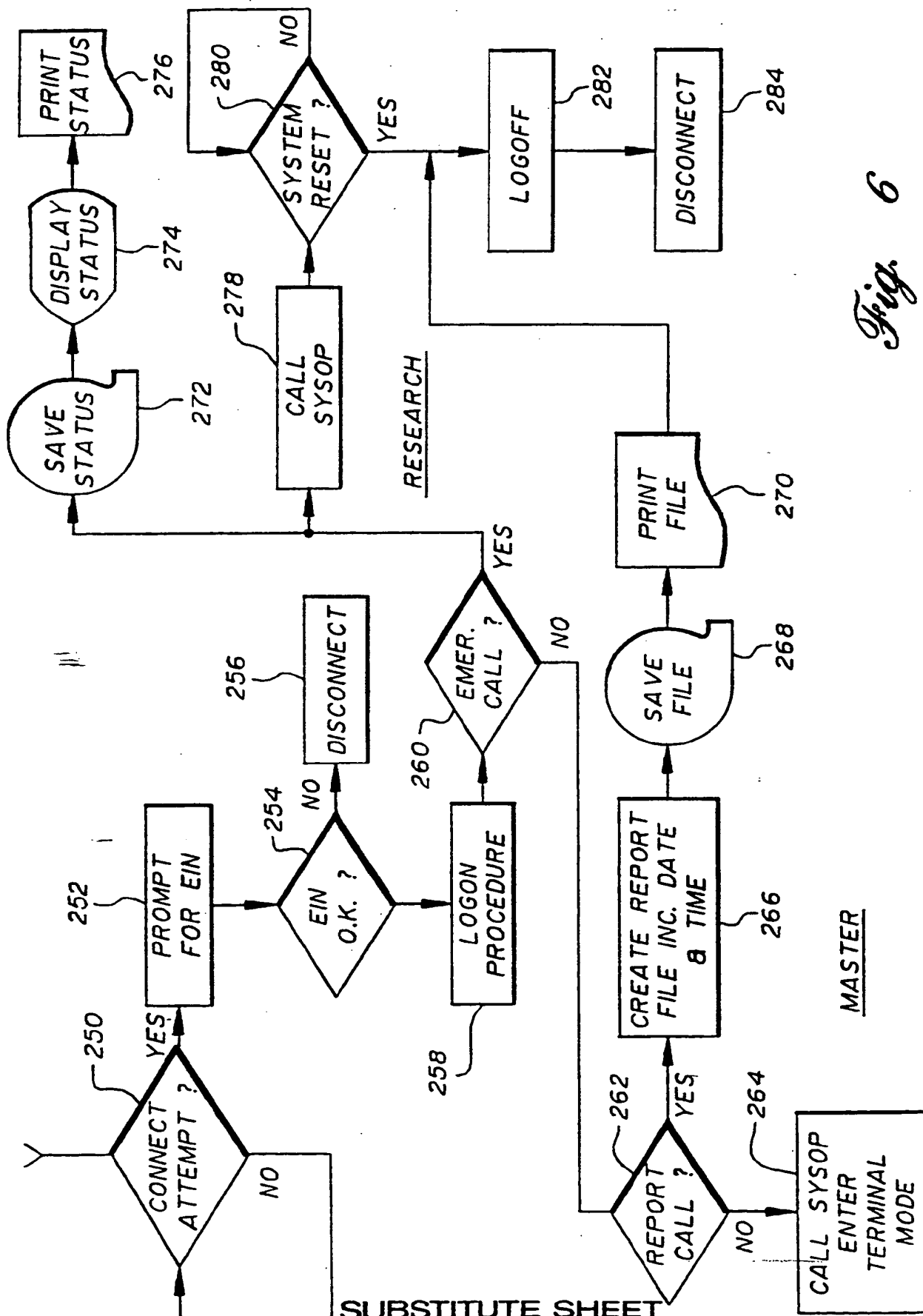


Fig. 5



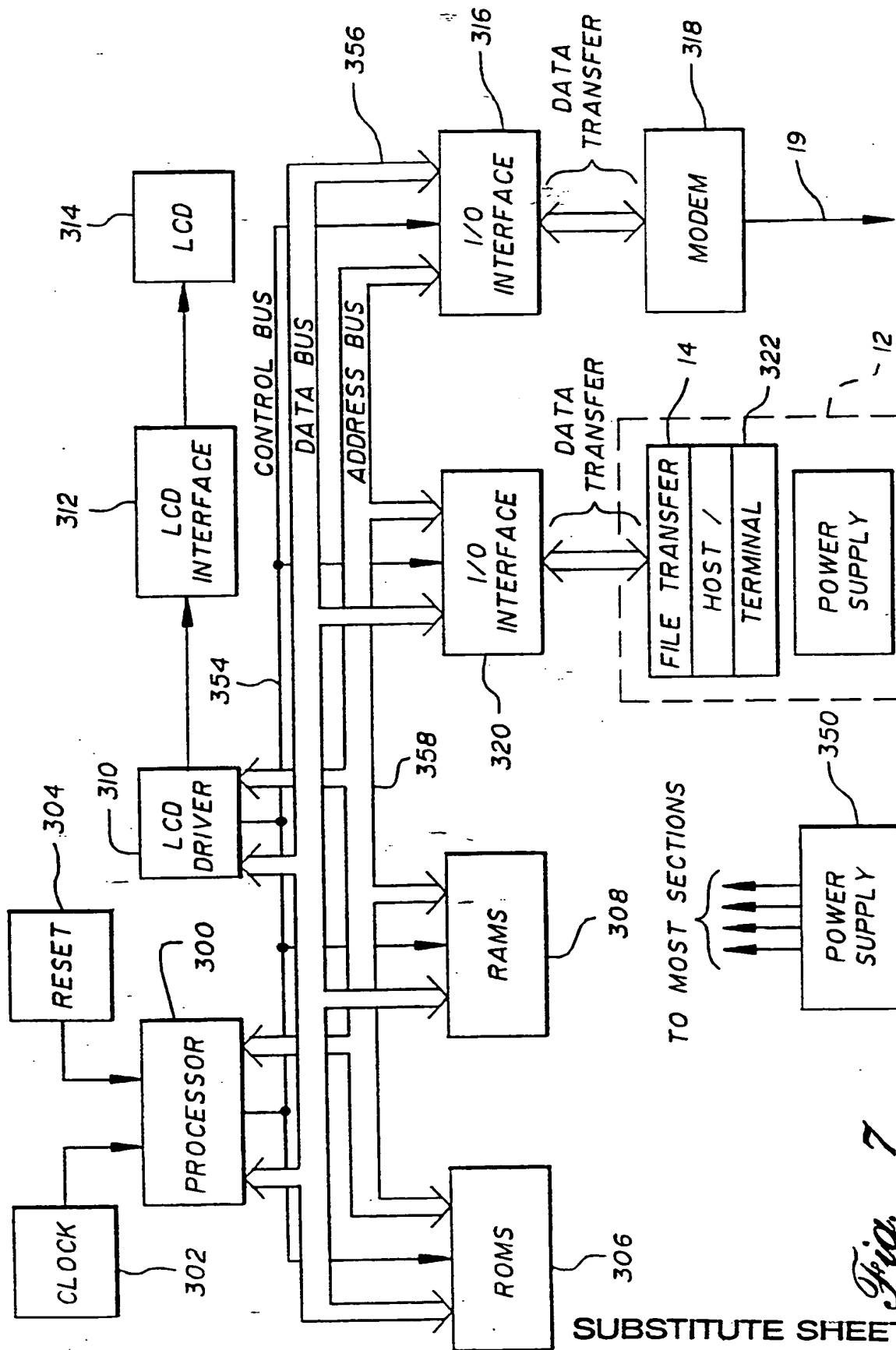


Fig. 7

SUBSTITUTE SHEET

BLOCK DIAGRAM OF SECURITY DEVICE

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/US93/05029

## A. CLASSIFICATION OF SUBJECT MATTER

IPC(5) :H04L 9/00

US CL :395/575

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 380/24,18,4,25; 371/14,19,67.1

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

AUTOMATED PATENT SEARCH:

Search terms: Virus and (computer? or program? or microprocessor? or software)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US, A, 4,975,950 (Lentz) 04 December 1990, see Abstract; Figures 1 and 2; and col. 1, lines 1 - col. 4, lines 51.	1-18
P,Y	US, A, 5,121,345 (Lentz) 09 June 1992, see Abstract; Figures 1 and 2; and col. 1, lines 1 - col. 4, line 60.	1-18
P,A	US, A, 5,144,660 (Rose) 01 September 1992, see Abstract; col. 1- col. 2, line 17.	1-18
P,A	US, A, 5,163,088 (LoCascio) 10 November 1992, see Abstract; Figures 1 and 2; and col. 4, lines 39048.	1-18

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:	*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
*A* document defining the general state of the art which is not considered to be part of particular relevance	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
*E* earlier document published on or after the international filing date	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*Z* document member of the same patent family
*O* document referring to an oral disclosure, use, exhibition or other means	
*P* document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

09 JULY 1993

Date of mailing of the international search report

21 SEP 1993

Name and mailing address of the ISA/US  
Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231

Authorized officer

LY V. HUA

Facsimile No. NOT APPLICABLE

Telephone No. (703) 305-9684